**Linux下的网桥简介**

编 写 人：孙自翔

统 稿 人：

编写时间：2014-03-17

部 门 名：

审 核 人：

审核时间：

修订记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 修订内容简述 | 修订日期 | 版本 | 修订人 |
| 1 | 初稿 | 2014-03-17 | 0.1 | 孙自翔 |
|  |  |  |  |  |
|  |  |  |  |  |

正文目录

[1 文档说明 4](#_Toc382820112)

[1.1 本文档目的 4](#_Toc382820113)

[1.2 本文档要达成的目标 4](#_Toc382820114)

[1.3 更进一步（最终目的） 4](#_Toc382820115)

[2 各种网络设备 4](#_Toc382820116)

[2.1 冲突域和广播域 4](#_Toc382820117)

[2.2 Hub和中继器 4](#_Toc382820118)

[2.3 交换机和网桥 4](#_Toc382820119)

[3 云计算环境下的网桥 4](#_Toc382820120)

[3.1 网络结构图 5](#_Toc382820121)

[4 linux下的bridge 5](#_Toc382820122)

[4.1 命令行创建网桥 5](#_Toc382820123)

[4.2 持久化配置 6](#_Toc382820124)

[4.3 bridge\_stp选项 6](#_Toc382820125)

[4.4 其他命令行选项 7](#_Toc382820126)

[4.5 常见故障诊断 8](#_Toc382820127)

[4.5.1 机器无法登陆 8](#_Toc382820128)

[4.5.2 添加接口失败 8](#_Toc382820129)

[5 vmware的三种网络连接方式 9](#_Toc382820130)

[6 云计算环境常见网络故障 9](#_Toc382820131)

[6.1 浮动IP ping不通 9](#_Toc382820132)

[6.2 无法获取固定IP 9](#_Toc382820133)

# 文档说明

## 本文档目的

云计算的网络环境比较复杂，其中涉及的概念比较多，某个概念理解不透彻就可能影响对整个环境的理解。

其中某些概念尤其重要，包括vlan、网桥。本文档对linux下网桥做一简单介绍，并介绍网易云计算环境中常见的网络故障及其排查方式.

## 本文档要达成的目标

* 理解网桥概念。
* Linux下如何创建网桥软设备。其中涉及到的vlan的概念，参考另一篇文档(linux下的vlan实践)
* 只对网桥做初步介绍以指导运维，某些概念（如stp）不做深入介绍。

## 更进一步（最终目的）

* 理解在各种情况下云网络的数据包走向.
* 云计算环境下网络故障排查。

# 各种网络设备

## 冲突域和广播域

<http://www.firewall.cx/networking-topics/general-networking/235-hubs-repeaters.html>

## Hub和中继器

<http://www.firewall.cx/networking-topics/general-networking/235-hubs-repeaters.html>

## 交换机和网桥

<http://www.firewall.cx/networking-topics/general-networking/236-switches-bridges.html>

# 云计算环境下的网桥

有了vlan和网桥的概念，就可以分析云计算环境下网络数据包的走向。

<http://www.mirantis.com/blog/openstack-networking-single-host-flatdhcpmanager/>

<http://www.mirantis.com/blog/openstack-networking-flatmanager-and-flatdhcpmanager/>

<http://www.mirantis.com/blog/openstack-networking-vlanmanager/>

## 网络结构图

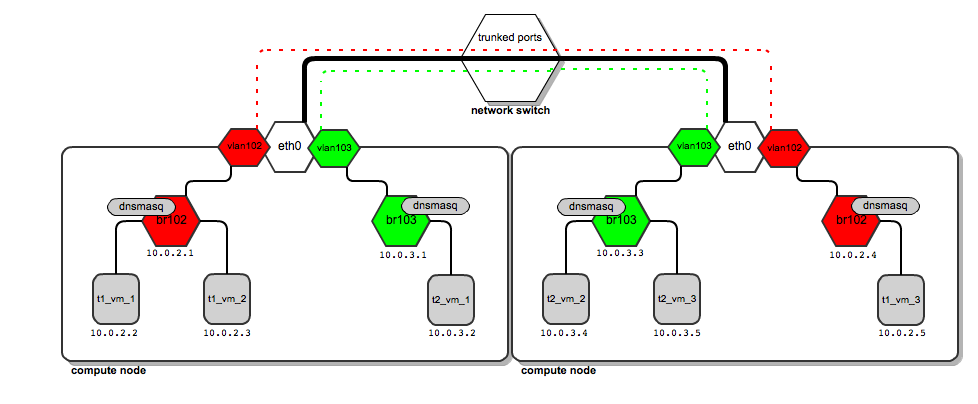


图4-1 云计算环境下的网络结构示意图

# linux下的bridge

## 命令行创建网桥

<http://www.microhowto.info/howto/bridge_traffic_between_two_or_more_ethernet_interfaces_on_linux.html>

* 依赖的模块和安装包

root@debian-1:~# lsmod |grep bridge

bridge 55884 0

stp 12405 2 bridge,garp

root@debian-1:~# dpkg -l|grep bridge-utils

ii bridge-utils 1.5-6 i386 Utilities for configuring the Linux Ethernet bridge

* 创建网桥

root@debian-1:~# brctl addbr br0

root@debian-1:~# brctl show br0

bridge name bridge id STP enabled interfaces

br0 8000.000000000000 no

root@debian-1:~# lsmod |grep bridge

bridge 55884 0

stp 12405 2 bridge,garp

root@debian-1:~# dpkg -l|grep bridge-utils

ii bridge-utils 1.5-6 i386 Utilities for configuring the Linux Ethernet bridge

* 开启STP

如果有需要可以开启STP,在云计算环境中，不需要开启这个选项。详见[4.3](#_bridge_stp选项)

root@debian-1:~# brctl stp br0 on

root@debian-1:~# brctl show br0

bridge name bridge id STP enabled interfaces

br0 8000.000000000000 yes

* 移除以太网接口的IP地址,原因参考[4.5.1](#_机器无法登陆)

IPV6地址会自动移除，IPV4不会

root@debian-1:~# ifconfig eth2 0.0.0.0 down

* attach以太网接口到网桥

可以添加指定的多个接口。

root@debian-1:~# brctl addif br0 eth2

root@debian-1:~# brctl show

bridge name bridge id STP enabled interfaces

br0 8000.000c2960a785 yes eth2

* 启动网桥

网桥是一个二层设备，建立好之后，跟其他因特网设备类似。

root@debian-1:~# ifconfig eth2 up

root@debian-1:~# ifconfig br0 up

* 配置IP

可以为网桥配置IP，以便对网桥进行管理，比如远程登录网桥.

root@debian-1:~# ifconfig br0 192.168.0.1

root@debian-1:~# ping 192.168.0.1 -c 1

PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.

64 bytes from 192.168.0.1: icmp\_req=1 ttl=64 time=0.041 ms

## 持久化配置

<http://www.faqs.org/docs/Linux-HOWTO/BRIDGE-STP-HOWTO.html>

auto br0

iface br0 inet manual

bridge\_ports eth0 eth1

bridge\_stp on

* 跟vlan不同，网桥不需要特殊的命名规则
* bridge\_ports关键字用来指示这是一个网桥。
* bridge\_stp关键字会影响网桥的启动时间, 详见[4.3](#_bridge_stp选项)。
* 同样可以把vlan设备添加进网桥。

同样可以为网桥配置IP。

auto br0

iface br0 inet static

bridge\_ports eth0 eth1

bridge\_stp on

address 192.168.0.2

netmask 255.255.255.0

gateway 192.168.0.1

## bridge\_stp选项

添加这个选项，会增加网桥启动时间。在云计算环境当中，网桥只是桥接一组虚拟机到一个物理接口，这个选项可以禁止。

<http://www.microhowto.info/howto/persistently_bridge_traffic_between_two_or_more_ethernet_interfaces_on_debian.html>

The bridge\_stp option specifies whether or not the Spanning Tree Protocol should be enabled. This is essential if there is any possibility of the bridge creating a loop in the network. It is safe in other cases, but it will increase the delay between a new link being added and it being able to pass traffic. For this reason you may want to leave STP disabled in simple cases (such as when bridging a set of virtual machines to a single physical interface).

实验如下：

root@debian-1:~# cat /etc/network/interfaces

auto br0

iface br0 inet static

bridge\_ports eth2

#bridge\_stp on

address 192.168.0.2

netmask 255.255.255.0

root@debian-1:~# ifup br0

Waiting for br0 to get ready (MAXWAIT is 32 seconds).

## 其他命令行选项

[http://iomem.com/index.php?archives/4-Ethernet-Bridges-under-Linux.html&serendipity[entrypage]=3&serendipity[entrypage]=all&serendipity[entrypage]=all&serendipity[entrypage]=all](http://iomem.com/index.php?archives/4-Ethernet-Bridges-under-Linux.html&serendipity%5bentrypage%5d=3&serendipity%5bentrypage%5d=all&serendipity%5bentrypage%5d=all&serendipity%5bentrypage%5d=all)

* 显示所有端口的mac地址

网桥的mac地址是port id最小的接口的mac地址

hzsunzixiang@10-120-120-15:~$ sudo brctl showmacs br100

* stp的各种参数

hzsunzixiang@10-120-120-15:~$ sudo brctl showstp br100

hzsunzixiang@10-120-120-15:~$ sudo brctl showmacs br100

* stp的各种参数调整

hzsunzixiang@10-120-120-15:~$ sudo brctl

Usage: brctl [commands]

commands:

addbr <bridge> add bridge

delbr <bridge> delete bridge

addif <bridge> <device> add interface to bridge

delif <bridge> <device> delete interface from bridge

hairpin <bridge> <port> {on|off} turn hairpin on/off

setageing <bridge> <time> set ageing time

setbridgeprio <bridge> <prio> set bridge priority

setfd <bridge> <time> set bridge forward delay

sethello <bridge> <time> set hello time

setmaxage <bridge> <time> set max message age

setpathcost <bridge> <port> <cost> set path cost

setportprio <bridge> <port> <prio> set port priority

show [ <bridge> ] show a list of bridges

showmacs <bridge> show a list of mac addrs

showstp <bridge> show bridge stp info

stp <bridge> {on|off} turn stp on/off

## 常见故障诊断

<http://www.microhowto.info/troubleshooting/troubleshooting_ethernet_bridging_on_linux.html>

### 机器无法登陆

一旦接口加入到一个网桥，这个接口就不能作为其他用途。特别是不能作为网络互联的终端。一个接口在加入网桥之后，需要先移除其ip,也不要在启动脚本中对这个接口运行dhcp。

如果此接口之前作为ssh等用途，加入网桥之后，就无法从这个接口登陆。

<http://www.microhowto.info/howto/bridge_traffic_between_two_or_more_ethernet_interfaces_on_linux.html>

Once an interface has been attached to a bridge it cannot be used for other purposes. In particular it cannot be used as an endpoint for Internet protocol traffic, so if the interface has been bound to any IP addresses then those addresses should be removed before the interface is attached to a bridge. If they are not removed then spurious entries will be left in the routing table which can disrupt connectivity.

实验步骤如下:

root@debian-1:~# cat /etc/network/interfaces

………..

auto eth2

iface eth2 inet static

address 192.168.2.201

netmask 255.255.255.0

auto br2

iface br2 inet static

bridge\_ports eth2

address 192.168.2.2

netmask 255.255.255.0

…………

root@debian-1:~# ifup eth2 # 启动eth2

root@debian-1:~# ping 192.168.2.201 -c 1 # 正常ping通

PING 192.168.2.201 (192.168.2.201) 56(84) bytes of data.

64 bytes from 192.168.2.201: icmp\_req=1 ttl=64 time=0.059 ms

--- 192.168.2.201 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 0.059/0.059/0.059/0.000 ms

root@debian-1:~# ifup br2 #启动网桥

Waiting for br2 to get ready (MAXWAIT is 32 seconds).

root@debian-1:~# brctl show br2　　　　　　　　　 #接口eth2加入网桥

bridge name bridge id STP enabled interfaces

br2 8000.000c2960a785 no eth2

root@debian-1:~# ping 192.168.2.201 -c 1 #此时再ping就会失败

PING 192.168.2.201 (192.168.2.201) 56(84) bytes of data.

From 192.168.2.2 icmp\_seq=1 Destination Host Unreachable

--- 192.168.2.201 ping statistics ---

1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

### 添加接口失败

一个接口不能同时隶属于两个网桥,否则会报错。

root@debian-1:~# ifup br2

device eth2 is already a member of a bridge; can't enslave it to bridge br2.

Waiting for br2 to get ready (MAXWAIT is 32 seconds).

# vmware的三种网络连接方式

理解了网桥的概念之后，可以很好的理解vmware的三种网络连接方式.

<http://vmfaq.com/entry/34/>

<http://www.huangkeye.cn/linux/593.html>

# 云计算环境常见网络故障

云计算环境中登陆VNC方便调试。

## 浮动IP ping不通

* 安全组规则是否设置。
* 是否获取了固定IP。可以进入vnc调试。
* 根据虚拟机的MAC地址查找宿主机上网桥对应的端口，是否添加进了网桥。

hzsunzixiang@10-120-120-15:~$ sudo brctl showmacs br100 |grep fa:16:3e:14:54:37

## 无法获取固定IP

* dnsmasq服务时否启动

hzsunzixiang@10-120-120-15:~$ ps aux|grep dnsmasq|grep -v grep

dnsmasq服务监听在网桥的IP上。

hzsunzixiang@10-120-120-15:~$ sudo netstat -apnt|grep dnsmasq

tcp 0 0 10.160.30.15:53 0.0.0.0:\* LISTEN 24614/dnsmasq

hzsunzixiang@10-120-120-15:~$ ip a s br100

7: br100: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc noqueue state UP

link/ether 00:e0:81:dd:e0:ea brd ff:ff:ff:ff:ff:ff

inet 10.160.30.15/23 brd 10.160.31.255 scope global br100

valid\_lft forever preferred\_lft forever

* iptables规则是否缺失

hzsunzixiang@10-120-120-15:~$ sudo iptables-save |grep CHECKSUM

-A POSTROUTING -p udp -m udp --dport 68 -j CHECKSUM --checksum-fill

-A nova-network-POSTROUTING -o br100 -p udp -m udp --dport 68 -j CHECKSUM --checksum-fill